International Journal of Scientific Research & Engineering Trends
Volume 6, Issue 3, May-June-2020, ISSN (Online): 2395-566X

Pre Migration Process for Scheduling Tasks in Cloud

Assistant Professor (Sl.Gr.),M.MohanaSundari, V.Adithyan,S.Madhumitha
Department of Computer Science and Engineering
Velalar College of Engineering and Technology
Erode, Thindal-12

***Abstract* – Cloud computing is the most recent technology in the computing field that makes network access easy. Many Cloud services for mapping Cloud software using virtualization techniques. The users' requests from various resources will cause a traffic problem at data centers. Thus, a Cloud load balancer that uses an efficient load balancing algorithm. Load balancing is an important factor in cloud computing since users accessing the services is increasing drastically day by day. The cloud is comprised of datacenters since it is very powerful to manage more number of users. But load balancing is a technique of distributing the workloads among different nodes to minimize response time, cost, resource utilization, and overheads. The aim of this project is to briefly analyze about various efficient load balancing algorithms and experimentally test how to minimize response as well as processing time through cloud analyst tool. The project proposes a new algorithm, weighted round robin, which optimizes pre-migration mechanism for load balancing of tasks among the multiple virtual machines in the Cloud server, thereby improving the response efficiency of Cloud servers. The methods and findings used to solve load imbalance issues in Cloud servers. The method used in this study will demonstrate that by applying an effective load balancing technique would improve the throughput and the response time in cloud environment. The main scope of the project is used to analyze and map the best cloud host for the selected virtual machine using Weighted Round Robin based algorithm. Cloud Sim 3.3 is used as the front end language to develop the application. Java 1.7 is used for coding.**

***Keywords*– Cloud computing, Load balancing, M-Learning (mobile learning), Virtual machines**

I. INTRODUCTION

Many computing methodologies are available in the computing field for maximizing automation. Among those, cloud computing is considered to be the best service oriented computing technologies to automate tasks in virtual machines as well as to enable users to access information very efficiently.

Also, m-learning offers cost effective solutions for a wide range of services. Cloud computing is an essential domain to explain distributed data sharing. In m-learning, mobile devices used by end users are called the m-learning clients. Through internet connectivity, m-learning clients store and retrieve data from Cloud data centers. Hence, systems integrated with Cloud data centers are quite advantageous for transferring all types of data and applications to system and mobile device easily and accurately. However, load balancing issues in Cloud data centers should be addressed to improve performance and efficiency.

Some of the Cloud computing services that could be used for learning approaches are Software-as-a Service (SaaS), Platform-as-a-Service (PaaS), Infrastructure as-a-Service (IaaS), and Hardware-as-a Service (HaaS).Load balancing techniques are used to distribute incoming traffic across multiple servers to minimize the delay of the Cloud server response to the Cloud users. Cloud load balancing is considered adequate only if the through put in the Cloud server is high, delays are minimal, and jitter is minimal while addressing Cloud user requests. Sometimes, failure occurs in load balancing in Cloud leads to poor image resolution and poor video streaming. So, load balancing in Cloud servers is essential to maximize throughput and to achieve superior performance in both public and private Clouds.

II. SCOPE AND OBJECTIVES

1. The dominant firefly behavior search model should be applied.
2. To balance the load in multiple Virtual Machines by increasing QoS metrics such as throughput and response time.
3. To demonstrate an improvement in energy consumption among Cloud servers.
4. To enhance m-learning environments by finding many relational models to avoid the highest energy consuming server throughout the world

III. EXISTING SYSTEM

The existing system consist of two level task scheduling mechanism to meet dynamic requirements of users. It obtains high resource utilization. This algorithm possesses load balancing by mapping tasks to virtual machines and all virtual machines to hosts. It is improving the task response time .It also provides better resource utilization. Round Robin algorithm is used here in which all the processes are divided between all processors. Each process is assigned to the individual processor in a round robin manner.

The load distributions between processors are equal. Different processes does not have same processing time. In many situations some nodes may be heavily loaded and others remain idle in servers where http requests are of similar nature and distributed equally then RR algorithm is used. In Round Robin Scheduling the time quantum play an important role. When time quantum is very large then RR When time quantum is too small then Round Robin Scheduling is known as Processor Sharing Algorithm.

1. Drawbacks of Existing System

- Round-robin scheduling algorithm is used in which a weight is assigned to each of the available cloud server nodes during the task execution time only.
- Average response time and Data center service time cannot be calculated since they are processed during the execution time only.
- Total cost of different data centers cannot be taken into account before migration status.

IV. PROPOSED SYSTEM

In existing system, a load balancing algorithm is proposed based on the availability of the VM before migration starts. Specifically, the Availability Index (AI) value is evaluated for every VM over a given period of time which will act as Weighted Round Robin, and therefore a task is assigned to the machine based on their Availability Index value. In order to validating the proposed model, it is compared with efficient load balancing algorithms, namely Round Robin, Throttled and Active Monitoring.

1. Advantages of Proposed System

- WRR is applied pre-migration state which is a modified version of the round-robin scheduling algorithm in which a weight is assigned to each of the available cloud server nodes.
- Average response time and Data center service time can be calculated prior to task execution starting time.
- Total cost of different data centers have been taken into account before migration status.

V. MODULE DESCRIPTION

1. Round Robin Load Balancing

In this module, round robin load balancing is used. The name of round robin algorithm suggests that it works in round robin manner. If the Data Center Controller receives a request from client it intimates the RR load balancer to allocate the new virtual machine for the purpose of processing. Round robin load balancer (RRLB) picked up the virtual machine randomly and returns the virtual machine id to Data Center Controller. In this way the frequent requests are processed in the circular order. This better allocation method is called weighted RR balancer in which we can assign a weight to each virtual machine so that one virtual machine is capable of handling.

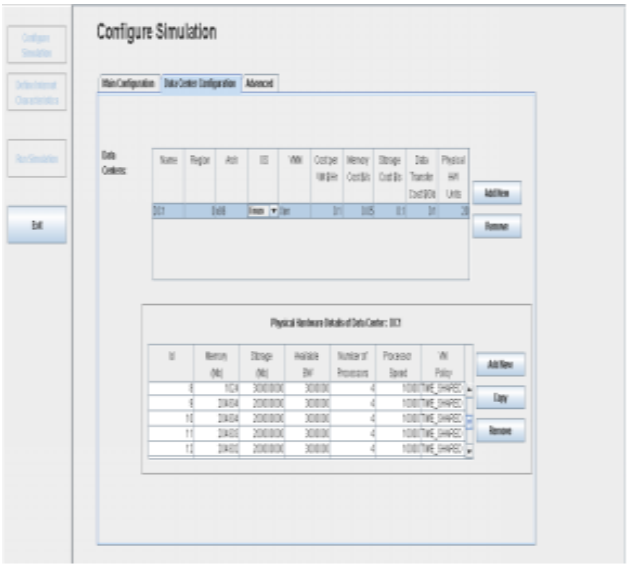

Fig. 1 Data center configuration

2. Active Monitoring Load Balancing

In this module, active monitoring load balancing is used. This is also known as equally spread the execution load balancing. It applies the active monitoring balancer to equally spreading execution of loads on the different virtual machines. The steps of this algorithm are described as follows referring to Fig. The term Active monitoring load balancer (AMLB) maintains an index table of virtual machines and the number of allocations assigned to each virtual machine.

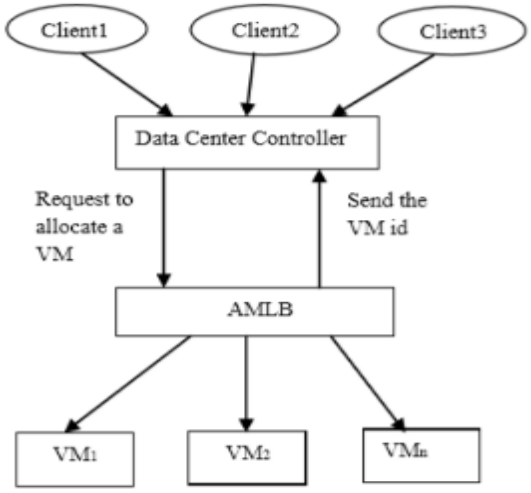

Fig. 3Active monitoring load balancing.

When a Data Center Controller receives a new request from the client. When the request for allocating the new virtual machine from Data Center Controller that arrives at AMLB, it parses an index table from top itself until the least loaded virtual machine is found. When it finds, it will returns the virtual machine id to Data Center Controller. If more than one virtual machine was found, AMLB possess first come first serve (FCFS) manner to choose the least loaded virtual machine.

Simultaneously, it returns the virtual machine id to Data Center Controller. Then the Data Center interacts with the virtual machine identified by that id. Data Center Controller finds the AMLB about new allocation. After the AMLB updates the allocation table by using increasing the allocation count by 1 for particular virtual machine. When virtual machine suitably finished the processing of the assigned request, it forwards the response to Data Center Controller. On receiving that response it intimates the AMLB about virtual machine re-allocation. This AMLB updates allocation table by decreases the allocation count for VM by 1.

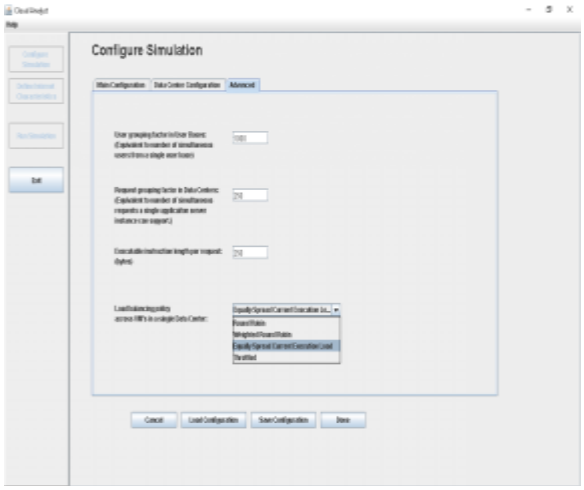

Fig. 4 Configure simulation form.

3. Throttled Load Balancing

In throttled load balancer (TLB) to monitoring the loads on each virtual machine. Each virtual machine is assigned to only one task at the time and can assigned other tasks when only the current task has executed successfully. The algorithm can be described as: The job of Throttled Load balancer maintains the index table of all the virtual machines as well as its current states (Available/Busy). First the client makes the request to Data Centre Controller for allocating the appropriate virtual machine and to perform the recommended job. Data Centre Controller notifies the TLB for allocating of the virtual machine. TLB scans the index table by top to bottom till the first available virtual machine is found.

If it found, then the TLB returns the id of the virtual machine to the Data Center Controller. The Data Centre interacts with the request to the virtual machine identified by its id. Afterwards, Data Centre acknowledges the TLB about new allocation of the virtual machine and revises an index table by increases the allocation of that virtual machine by 1.

Otherwise, if the TLB did not find any virtual machine in the active state it simply returns nothing. In this scenario Data Center Controller makes the queue for the request till the availability of the virtual machine. When the virtual machine finishes the processing of the request, it passes a response to Data Center Controller. On receiving that, Data Center Controller gives the acknowledgement about the TLB regarding virtual machine re-allocation. The TLB updates allocation table by decreases the allocation count of the virtual machine by 1.

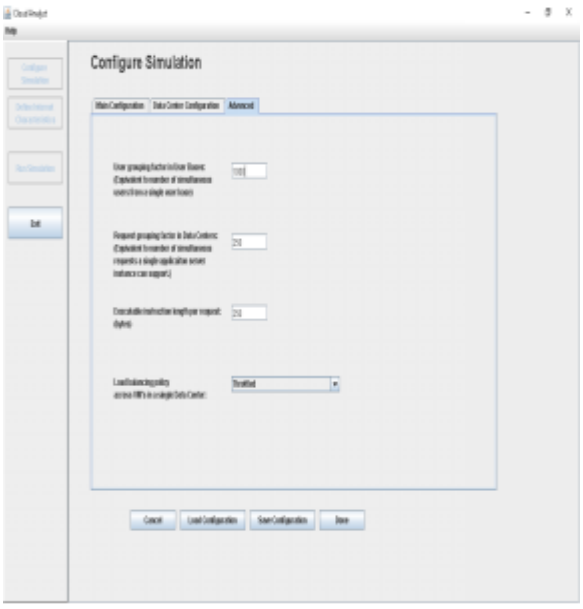

Fig.5 Load balancing policy setting.

4. Weighted Round Robin Load Balancing

The name of weighted round robin algorithm suggests that it works in round robin manner. When the Data Center

Controller gets the request from the client it intimates the weighted RR load balancer for to allocate the new virtual machine for the processing. WRRLB picks a virtual machine randomly and returns the VM id to Data Center Controller for processing. The cloud node with maximum capability to create and map the VM is assigned to the selected virtual machine. It is assumed that all the Virtual machines with same requirements can be mapped to this particular cloud host node.

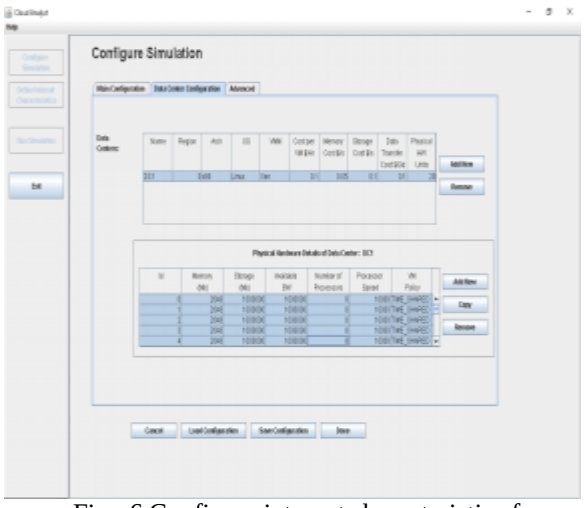

Fig. 6 Configure internet characteristics form

VI. CONCLUSION

Response Time minimization is a greater challenge widely seen in each and every aspect of IT sector for developing the products/applications which increase the efficiency of business performance and thereby customer satisfaction in cloud based environment. Keeping these things in mind this study analyzes the four major load balancing algorithms: Round robin, Weighted Round robin, Active monitoring and Throttled.

Keeping the number of processors of each VM same it is found that Weighted Round robin load balancing as the efficient one. But sometimes it may not be possible for that each data centre has similar number of processors per virtual machine. On the other hand it can help the professionals undoubtedly. It is also found the parameters: processing time, response time of the process are almost similar in both scenario of active monitoring as well as throttled load balancing.

Whatever, these parameters are improved in case of active monitoring balancing. Hence it is concluded that active monitoring load balancing is an effective and efficient one than the other two that are discussed. When four algorithms are compared Weighted Round robin becomes the most effective one in both response time and processing time. Moreover, during weighted round robin based algorithm testing, case 1 is used with 4 processors allocation for all virtual machines and case 2 is used with 8 processors for 10 virtual machines and 4 processors for other 10 virtual machines. It is observed that very small improvement is made in processing time. At present, during Virtual Machine failure during allocation to host, the maximum capability host is allocated to that virtual machine during the simulation. In future, all the virtual machines can be mapped in the order or maximum capability hosts.

REFERENCES

[1]. M.ShreedharandG.Varghese,‘‘Efficient fair queuing using deficit round robin,’’ IEEE/ACM Trans. Netw., vol. 4, no. 3, pp. 375–385, Jun. 1996

[2]. C.Fehling,T.Ewald,F.Leymann,M.Pauly,J.Rütschlin,an dD.Schumm, ‘‘Capturing cloud computing knowledge and experience in patterns,’’ in Proc. IEEE 5th Int. Conf. Cloud Comput. (CLOUD), Jun. 2012, pp. 726–733.

[3]. L. D. D. Babu and P. V. Krishna, ‘‘Honey bee behavior inspired load balancing of tasks in cloud computing environments,’’Appl.Soft Comput., vol. 13, no. 5, pp. 2292–2303, May 2013.

[4]. B.Mondal, K.Dasgupta, and P.Dutta, “Load balancing in Cloud computing using stochastic hill climbing-a soft computing approach,” Procedia Technol., vol. 4, pp. 783–789, Jun. 2012.

[5]. T. R.Armstrong, D.Hensgen, The relative performance of various mapping algorithms is independent of sizable variances in runtime predictions, in: 7th IEEE Heterogeneous Computing Workshop (HCW ’98), 1998, pp. 79–87.

[6]. Vouk, Cloud computing- issues, research and implementations, in: Information Technology Interfaces, 2008, pp. 31–40.